

Задача А. Словарь

Имя входного файла: `dictionary.in`
Имя выходного файла: `dictionary.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дан набор слов и текст, требуется определить для каждого слова, присутствует ли оно в тексте как подстрока.

Формат входных данных

В первой строке дан текст (не более 10^6 строчных латинских букв). Далее дано число M — количество слов в словаре.

В следующих M строках записаны слова (не более 30 строчных латинских букв). Слова различны и отсортированы в лексикографическом порядке.

Суммарная длина слов в словаре не более 10^5 .

Формат выходных данных

M строк вида `Yes`, если слово присутствует, и `No` иначе.

Пример

<code>dictionary.in</code>	<code>dictionary.out</code>
<code>trololo</code>	<code>No</code>
<code>3</code>	<code>Yes</code>
<code>abacabadabacaba</code>	<code>Yes</code>
<code>olo</code>	
<code>trol</code>	

Задача В. Префикс-функция

Имя входного файла: `prefix.in`
Имя выходного файла: `prefix.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Постройте префикс-функцию для заданной строки s .

Формат входных данных

Первая строка входного файла содержит s ($1 \leq |s| \leq 10^6$). Строка состоит из букв латинского алфавита.

Формат выходных данных

Выведите значения префикс-функции строки s для всех индексов $1, 2, \dots, |s|$.

Пример

<code>prefix.in</code>	<code>prefix.out</code>
aaaAAA	0 1 2 0 0 0

Задача С. Z-функция

Имя входного файла: `z.in`
Имя выходного файла: `z.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Постройте Z-функцию для заданной строки s .

Формат входных данных

Первая строка входного файла содержит s ($1 \leq |s| \leq 10^6$). Строка состоит из букв латинского алфавита.

Формат выходных данных

Выведите значения Z-функции строки s для индексов $2, 3, \dots, |s|$.

Примеры

<code>z.in</code>	<code>z.out</code>
aaaAAA	2 1 0 0 0
abacaba	0 1 0 3 0 1

Задача D. Коровы бега

Имя входного файла: `cowrun.in`
Имя выходного файла: `cowrun.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Фермер Джон забыл заделать дыру в изгороди на своей ферме и его n коров сбежали и бедокурят. Каждая минута, когда корова находится вне изгороди, она "бедокурит" на 1 доллар. ФД должен посетить каждую корову, чтобы "усмирить" ее и прекратить долларовые потери от нее.

К счастью, коровы находятся вдоль одной прямой на различных расстояниях от фермы. ФД знает расстояние p_i для каждой коровы i относительно ворот (позиция 0), из которых он стартует.

ФД движется на 1 единицу расстояния за минуту и усаживает корову мгновенно. Определите порядок, в котором ФД должен посещать коров, так чтобы минимизировать свои долларовые потери.

Формат входных данных

Первая строка содержит число n — количество коров ($1 \leq n \leq 1000$). Следующие n содержат p_i — расстояние от i -ой коровы до ворот ($-500\,000 \leq p_i \leq 500\,000, p_i \neq 0$).

Формат выходных данных

Выведите минимальную общую стоимость долларовых потерь.

Пример

cowrun.in	cowrun.out
4 -2 -12 3 7	50

Задача Е. Прогулка по холмам

Имя входного файла: `hillwalk.in`
Имя выходного файла: `hillwalk.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Имеется n холмов. Каждый холм имеет форму отрезка из точки (x_1, y_1) в точку (x_2, y_2) где $x_1 < x_2$ и $y_1 < y_2$. Никакие из этих отрезков не пересекаются и не касаются даже в конечных точках. Кроме того, для первого холма справедливо $(x_1, y_1) = (0, 0)$.

Беси начинает свой путь в точке $(0, 0)$ на первом холме. Когда Беси попадает на холм, она карабкается вверх пока не достигнет конца холма. Затем она прыгает вниз. Если она приземлится на другой холм, она продолжит карабкание уже на этом холме, иначе она падает в бездну (где $y = -\infty$). Каждый холм $(x_1, y_1) \rightarrow (x_2, y_2)$ необходимо рассматривать как содержащий точку (x_1, y_1) , но не содержащий точку (x_2, y_2) , поэтому Беси приземляется на холм, если она падает на него сверху с позиции $x = x_1$, но не приземлится на него, если она падает сверху с позиции $x = x_2$.

Посчитайте общее количество холмов, которых Беси коснется в некоторой точке во время своего путешествия.

Формат входных данных

В первой строке задано натуральное число n — количество холмов ($1 \leq n \leq 100,000$).

Следующие n строк содержат по четыре целых числа (x_1, y_1, x_2, y_2) , описывающих холм i ($0 \leq x_1, y_1, x_2, y_2 \leq 10^9$).

Формат выходных данных

Требуется вывести единственное целое — количество холмов, которых коснется Беси за время своего путешествия.

Пример

hillwalk.in	hillwalk.out
4 0 0 5 6 1 0 2 1 7 2 8 5 3 0 7 7	3

Замечание

Имеется 4 холма. Первый холм из $(0,0)$ в $(5,6)$ и т.д.

Беси пройдет по холмам #1, #4, #3.

Задача F. Necklace

Имя входного файла: `necklace.in`
Имя выходного файла: `necklace.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Беси выложила n камней, на каждом одна буква алфавита и хочет построить ожерелье.

Имя соседки Беси представляет строку из m символов. Беси хочет, чтобы эта строка из m символов не встречалась как непрерывная подстрока в строке, представляющей ее ожерелье.

Беси решила удалить некоторые из камней из своего ожерелья, так чтобы имя другой коровы не встречалась как подстрока.

Определите минимальное количество камней, которое она должна удалить.

Формат входных данных

Строка длины n , описывающая ожерелье Беси все символы в диапазоне от a до z . Строка длины m , описывающая имя другой коровы все символы в диапазоне от a до z .

Формат выходных данных

Минимальное количество камней, которое нужно удалить из ожерелья Беси, чтобы оно не содержало имя другой коровы как подстроку.

Пример

<code>necklace.in</code>	<code>necklace.out</code>
ababaa aba	1

Задача G. Задача для Ильдара

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 4 секунды
Ограничение по памяти: 512 мегабайт

Вам дан ориентированный ациклический граф, состоящий из n вершин и m ребер. Каждая вершина v имеет значение a_v , изначально все $a_v = 0$. Заданы q запросов следующего типа.

- 1 u x . Присвоить значение x ко всем вершинам v , таким, что существует путь из u в v .
- 2 u x . Присвоить значение x ко всем вершинам v , таким, что существует путь из u в v и $a_v > x$.
- 3 u . Вывести значение a_u .

Формат входных данных

Первая строка содержит три числа n, m, q — количество вершин, ребер и запросов соответственно ($2 \leq n \leq 10^5, 1 \leq m, q \leq 10^5$). Следующие m строк содержат числа u и v ($1 \leq u, v \leq n, u \neq v$), обозначающие, что есть ребро из u в v . Следующие q строк содержат запросы в формате, описанном выше ($0 \leq x \leq 10^9$).

Формат выходных данных

Для каждого запроса третьего типа выведите ответ на запрос.

Пример

стандартный ввод	стандартный вывод
6 5 18	3
1 2	3
1 3	3
3 4	3
2 4	3
5 6	2
1 1 3	3
3 1	2
3 2	0
3 3	0
3 4	3
1 2 2	2
3 1	3
3 2	2
3 3	
3 4	
2 6 7	
3 5	
3 6	
2 1 3	
3 1	
3 2	
3 3	
3 4	